function instance when the safety margin area comprises the complete network domain. At the design time the function designer can for example define the required impact area for the PCI function, as for example third degree neighbours.

[0197] The generic coordination logic determiner in this example can be configured to determine coordination logic with respect to the PCI/PCI case (in other words when two instances of type PCI) are considered. An example coordination logic would be that a first instance appearing at the coordination layer gets priority over a second one appearing later.

[0198] The generic impact time determiner can be configured for this example to set the impact time to be the actual time required to configure a PCI on the NE (for example including the time required to "reboot" a cell). In some embodiments the impact time can also comprise a operator-set minimum time interval between two PCI changes.

[0199] When at run-time a 'first' PCI function instance execution request is received, the coordination layer and in some embodiments the event processor **331** identifies the target cell as the function area and the actual impact determiner **313** derives the impact area in relation to this cell. As this is the 'first' PCI function instance there are no conflicts and the coordination determiner **315** outputs an acknowledgement. This information together with the maximal impact time is stored within the context information by the context determiner **317** performing a context update operation. Furthermore the function instance can be executed.

[0200] When another PCI function instance execution request is received by the coordination layer the information provided in the request event by the event processor is used by the actual impact determiner **313** to derive the impact area.

[0201] The coordination determiner is then configured to evaluate the context information to determiner whether there are other potentially conflicting function instances with a shared impact area. If this is the case the new function execution request is treated according to the coordination logic (for example the other request is terminated).

[0202] It would be understood that the embodiments of the application allow a preferable safety vs. efficiency tradeoff which can be governed by adjusting impact area and impact time parameters (and for example provide a safety margin area such as shown in the above example).

[0203] Furthermore in some embodiments of the application only minimal limitations on SON function design are imposed, in that events which are related potentially to other function instances need to be exposed to the coordination layer (coordination and the associated knowledge resides separately in the coordination layer rather than in the individual functions). Therefore embodiments of the application can be applied to both centralized and distributed SON function instances as long as the request events are exposed to the coordination layer (for centralized functions in the form of local events, for distributed functions as events based on a network protocol). Furthermore it can be possible in some embodiments of the application to avoid deadlocks, race conditions, oscillations by implementing an active coordination of SON functions. Furthermore these embodiments permit flexible coordination by amendment of the coordination logic with a rule engine allowing the system at run-time to adapt. Thus in some embodiments the "design-time" is not strictly the time before the whole system goes into operation, but before new functionality based on a new/updated rule is put into operation.

[0204] It is also noted herein that while the above describes exemplifying embodiments of the invention, there are several variations and modifications which may be made to the disclosed solution without departing from the scope of the present invention.

[0205] In general, the various embodiments may be implemented in hardware or special purpose circuits, software, logic or any combination thereof. Some aspects of the invention may be implemented in hardware, while other aspects may be implemented in firmware or software which may be executed by a controller, microprocessor or other computing device, although the invention is not limited thereto. While various aspects of the invention may be illustrated and described as block diagrams, flow charts, or using some other pictorial representation, it is well understood that these blocks, apparatus, systems, techniques or methods described herein may be implemented in, as non-limiting examples, hardware, software, firmware, special purpose circuits or logic, general purpose hardware or controller or other computing devices, or some combination thereof.

[0206] The embodiments of this invention may be implemented by computer software executable by a data processor of the mobile device, such as in the processor entity, or by hardware, or by a combination of software and hardware.

[0207] Further in this regard it should be noted that any blocks of the logic flow as in the Figures may represent program steps, or interconnected logic circuits, blocks and functions, or a combination of program steps and logic circuits, blocks and functions. The software may be stored on such physical media as memory chips, or memory blocks implemented within the processor, magnetic media such as hard disk or floppy disks, and optical media such as for example DVD and the data variants thereof, CD.

[0208] The memory may be of any type suitable to the local technical environment and may be implemented using any suitable data storage technology, such as semiconductor-based memory devices, magnetic memory devices and systems, optical memory devices and systems, fixed memory and removable memory.

[0209] The foregoing description has provided by way of exemplary and non-limiting examples a full and informative description of the exemplary embodiment of this invention. However, various modifications and adaptations may become apparent to those skilled in the relevant arts in view of the foregoing description, when read in conjunction with the accompanying drawings and the appended claims. However, all such and similar modifications of the teachings of this invention will still fall within the scope of this invention as defined in the appended claims. Indeed, there is a further embodiment comprising a combination of one or more of any of the other embodiments previously discussed.

1. A method of self-organizing a telecommunications network, the method comprising:

determining at least one event criteria from a received event function request;

determining an event impact from the at least one event criteria and a general function impact;

determining whether the event impact interferes with at least one implemented function instance; and

generating a coordination output based on the determination whether the event impact interferes.

2. The method as claimed in claim **1**, wherein determining at least one event criteria comprises determining at least one of: